Experiment 3: Link Layer

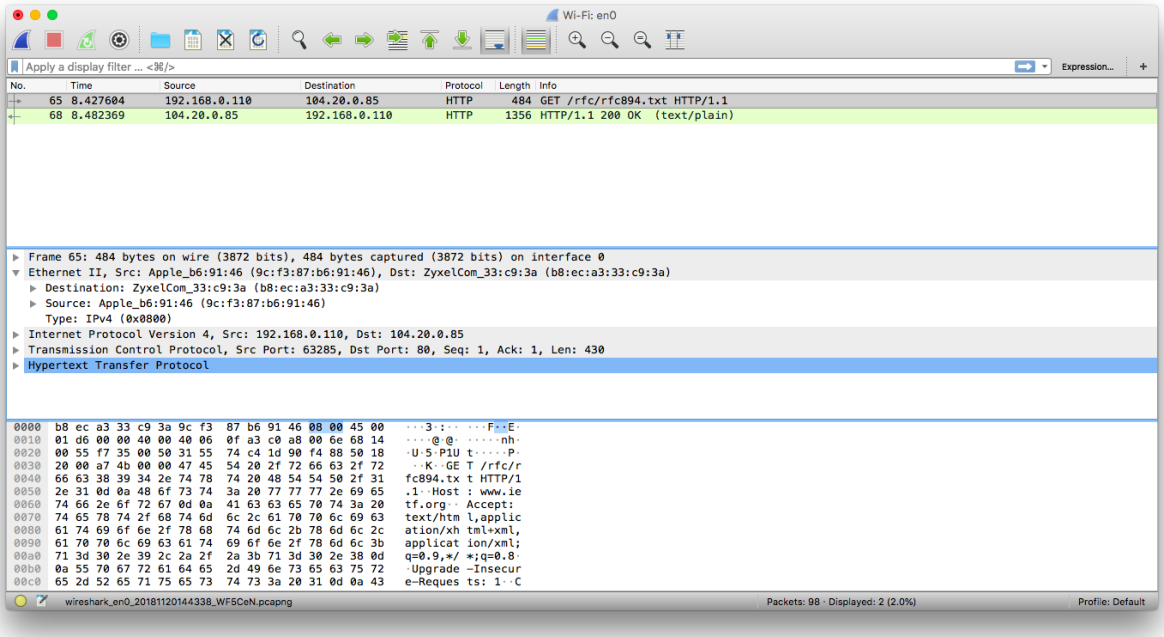
1 Introduction

In this experiment, you will explore protocols operating at the link layer. We haven't studied the Ethernet and Wireless protocols in detail yet, but I've uploaded the lecture notes (Lecture 16) so you can consult these to learn more.

2 Ethernet/Wireless Frames

Capture a set of Ethernet frames to study via an Ethernet or WiFi connection.

1. Make sure your browser's cache is empty.
2. Start the Wireshark packet sniffer.
3. Enter the following URL into your browser http://www.ietf.org/rfc/rfc894.txt
4. Your browser should display the RFC for IP over Ethernet.
5. Stop the Wireshark packet capture. First, find the packet numbers (the leftmost column in the upper Wireshark window) of the HTTP GET message that was sent from your computer to www.ietf.org, as well as the beginning of the HTTP response message sent to your computer by www.ietf.org. You should see a screen that looks something like this (where packet 65 in the screen shot below contains the HTTP GET message).



In order to answer the following questions, you'll need to look into the packet details and packet contents windows (the middle and lower display windows in Wireshark).

Include the answers to the questions in your very brief report (and where appropriate explain how you obtained the solution). The report can just be a series of answers to the questions.

Select the Ethernet frame containing the HTTP GET message. (Recall that the HTTP GET message is carried inside of a TCP segment, which is carried inside of an IP datagram, which is carried inside of an Ethernet frame). Expand the Ethernet II information in the packet details window. Note that the contents of the Ethernet frame (header as well as payload) are displayed in the packet contents window.
Answer the following questions, based on the contents of the Ethernet frame containing the HTTP GET message.

2.1 What is the 48-bit Ethernet address of your computer? Where did you find this information?

2.2 What is the 48-bit destination address in the Ethernet frame? Is this the Ethernet address of www.ietf.org? (Hint: the answer is no). What device has this as its Ethernet address? Again, describe where you got this information.

2.3 Who is the manufacturer of the Ethernet adapter (or computer) for both the source and destination Ethernet adapters. Determine this from the Ethernet address, not by looking at the label on your laptop. Use the document at http://standards-oui.ieee.org/oui.txt - don't just rely on Wireshark, although you should see a match. An Organizationally Unique Identifier (OUI) is a 24-bit number that uniquely identifies a vendor or manufacturer – it is essentially the first three octets of a MAC address.

2.4 Give the hexadecimal value for the two-byte TYPE field. Interpret the meaning of this value.

2.5 How many bytes from the very start of the Ethernet frame does the ASCII "T" in "GET" appear in the Ethernet frame?

Next, answer the following questions, based on the contents of the Ethernet frame containing the first byte of the HTTP **response** message.

2.6 What is the value of the Ethernet source address? Is this the address of your computer, or of www.ietf.org? (Hint: the answer is no). What device has this as its Ethernet address?

2.7 What is the destination address in the Ethernet frame? Is this the Ethernet address of your computer?

3 Address Resolution Protocol (ARP)

In this section, you will observe ARP in action.

ARP Caching

Recall that the ARP protocol typically maintains a cache of IP-to-Ethernet address translation pairs on your computer. The *arp* command is used to view and manipulate the contents of this cache. Since the *arp* command and the ARP protocol have the same name, it's understandably easy to confuse them. But keep in mind that they are different

- the *arp* command is used to view and manipulate the ARP cache contents, while the ARP protocol defines the format and meaning of the messages sent and received, and defines the actions taken on message transmission and receipt.

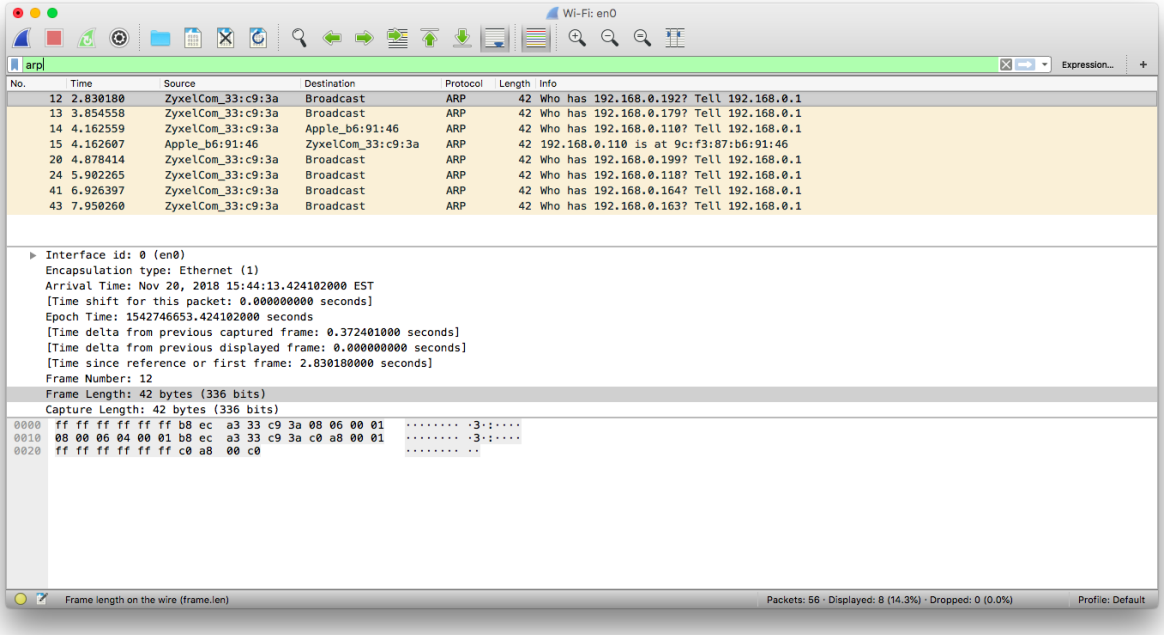
Let's take a look at the contents of the ARP cache on your computer. The *arp* command will display the contents of the ARP cache on your computer. For Windows1 and Mac OSX, use *arp -a*, for some variants of Linux, just use *arp*. Run the *arp* or *arp -a* command and keep a copy of the contents of your computer's ARP cache.

In order to observe your computer sending and receiving ARP messages, we'll need to clear the ARP cache, since otherwise your computer is likely to find a needed IP- Ethernet address translation pair in its cache and consequently not need to send out an ARP message. The -d flag to the arp command deletes an entry from the ARP cache.
Use *arp -ad* to clear all entries from the cache. You may need to look up exactly how to do this for your particular operating system. (Mac OS will need a sudo; Linux has the ip command – "ip -s -s neigh flush all".

Please do the following:

• Clear your ARP cache, as described above.
• Next, make sure your browser's cache is empty.
• Start up the Wireshark packet sniffer
• Enter the following URL into your browser http://www.ietf.org/rfc/rfc826.txt
• Your browser should display the RFC for ARP.
• Stop Wireshark packet capture. Filter on "arp" to isolate the ARP packets

Do the following:
You should now see a Wireshark window that looks like:



3.1 What are the hexadecimal values for the source and destination addresses in the Ethernet frame containing the ARP request message?

3.2 Give the hexadecimal value for the two-byte Ethernet Frame type field. How is this value different from the one you saw in Section 2?

3.3 You may need to refer to the ARP specification (in the RFC you just downloaded). A readable, detailed discussion of ARP is also at http://www.erg.abdn.ac.uk/users/gorry/course/inet-pages/arp.html. This is also a very useful page:
http://www.networksorcery.com/enp/protocol/arp.htm

a. How many bytes from the very beginning of the Ethernet frame does the ARP opcode field begin?
b. What is the value of the opcode field within the ARP-payload part of the Ethernet frame in which an ARP request is made? What does it mean?
c. Does the ARP message contain the IP address of the sender? Why or why not?
d. Where in the ARP request does the "question" appear – the Ethernet address of the machine whose corresponding IP address is being queried?

3.4 Now find the ARP reply that was sent in response to the ARP request.
a. How many bytes from the very beginning of the Ethernet frame does the ARP opcode field begin?
b. What is the value of the opcode field within the ARP-payload part of the Ethernet frame in which an ARP response is made? What does it mean?
c. Where in the ARP message does the "answer" to the earlier ARP request appear – the MAC address of the machine IP address is being queried?

3.5 What are the hexadecimal values for the source and destination addresses in the Ethernet frame containing the ARP reply message?

3.6 The arp command: (sudo) arp -s InetAddr EtherAddr allows you to manually add an entry to the ARP cache that resolves the IP address InetAddr to the physical address EtherAddr. What would happen if, when you manually added an entry, you entered the correct IP address, but the wrong Ethernet address for that remote interface? Try it and report on your findings.

3.7 What is the default amount of time that an entry remains in your ARP cache before being removed. You can determine this empirically (by monitoring the cache contents) or by looking this up in your operation system documentation. Indicate how/where you determined this value.

4 Wireshark Analysis

Open the Mystery.cap file in Wireshark.

4.1 For each of the packets, write a short description of what the *purpose* of the packet is. Back your assertion up with data from the packet. List anything else interesting in the packet. The goal is not to provide a recital of the basic protocol information. If the trace was a DHCP exchange, then there would be no need to write "This is a UDP packet sent to port 67." Instead I would expect something like "This is the DHCP OFFER message from the server. You can see the XID field is the same as the DHCP DISCOVER message…"

5 Important Dates and Evaluation

5.1 Demo

The demo will take place in class on Nov. 5 or 6 and will count for 2.5% of your final grade in the course. For the demo, you should be prepared to explain your answers to all the questions above and show the TA how you obtained your answers. You should be able to answer questions about the operation of the address resolution protocol and MAC addresses.

5.2 Report

The report is due at 23:59 on Nov. 9, and it will count for 2.5% of your final grade. The report should be nothing more than responses to all the questions in this document (expected length approximately 2 pages excluding figures). Please include a few screenshots from Wireshark to support your answers.